

# MATHEMATIK für die ASTRONOMIE

KULTURBUND DER DDR

Zentrale Kommission Astronomie und Raumfahrt – Arbeitskreis „Numerische Astronomie“

Veröffentlichung 6

Leipzig 1988

## Der Sonnenort nach NEWCOMB – ein Pascal-Programm

Für zahlreiche Anwendungen ist die Kenntnis des genauen Sonnenortes notwendig. So hängt z.B. der Erfolg einer Bahnbestimmung wesentlich von der Qualität dieses Ortes ab. Auch bei der Umrechnung von geozentrischen in heliozentrische Koordinaten bzw. umgekehrt geht der Fehler im Sonnenort in voller Höhe ein.

Die Genauigkeit einer mit den Formeln des Zweikörperproblems berechneten Ephemeride ist nur gering; deshalb soll hier die von NEWCOMB im Jahre 1898 veröffentlichte Theorie vorgestellt werden. In diesen Formeln werden die Einflüsse von Venus, Mars, Jupiter und Saturn sowie des Mondes berücksichtigt.

Das vorliegende Programm erfasst alle Störungen, die grösser als

und  $0.1$  Bogensekunden in Länge und Breite  
 $0.00000001$  Astronomische Einheiten im Radius

sind.

Folgende Genauigkeit wird erreicht:

Koordinate	typischer Fehler	Maximalfehler (Summe der Restterme)
Radius	0.000 002 AE	0.000 005 AE
Länge	1 "	2 "
Breite	0.2 "	0.5 "
Rektaszension	0.2 s	
Deklination	0.2 "	

Das Programm in der abgedruckten Form läuft ungeändert auf Turbo-Pascal 2.0..4.0, es werden jedoch keine Turbo-Spezifika benutzt. Falls ein anderer Compiler benutzt wird, dann sollte eine eventuell vorhandene höhere Genauigkeit für Gleitpunktzahlen benutzt werden. Die Konstanten der Störungsterme sind in der (Text)Datei SUNNEW.DAT abgelegt, die zur Laufzeit im Laufwerk A: erwartet wird.

### Beschreibung des Programms

Das Programm besteht im Wesentlichen aus zwei Teilen: Der Berechnung des Sonnenortes zum mittlere Äquinoktium des Datums (Theorie von NEWCOMB) und der Umwandlung mittlerer-scheinbarer Ort.

Das Programm stellt als erstes einige universell benutzbare Konstanten und Funktionen bereit.

Die Prozedur `sun_new_mitt` enthält die eigentliche Theorie des Sonnenortes. Die hiermit berechnete Sonnenlänge setzt sich aus

mittlerer Sonnenlänge	LO
Lösung der Mittelpunktsgleichung	DL
Störungen durch die grossen Planeten	DLx
Umrechnung Baryzentrum-Geozentrum (Mond)	DLM

zusammen.

Alle diese Werte werden in getrennten Funktionen/Prozeduren berechnet, wobei die Daten der Planeteneinflüsse aus den ersten vier Abschnitten der Datei entnommen werden.

Bei der Berechnung der benötigten mittleren Anomalie  $M$  werden langperiodische Störungen (DLP) berücksichtigt.

Der Logarithmus des Radiusvektors besteht ebenfalls aus zahlreichen Komponenten:

Wert aus der Mittelpunktsgleichung	RO
Störungen durch die Planeten	DRx
Einfluss des Mondes	RM

Die Konstanten der Störungsterme für Venus, Mars, Jupiter und Saturn sind in den Abschnitten 4..8 der Datei enthalten.

Die Sonnenbreite übersteigt niemals  $1.2''$ , deshalb kommt man bei ihrer Berechnung mit wenigen Werten aus. Die Werte für den Einfluss von Venus, Jupiter und Mond entstammen dem letzten Abschnitt der Datei.

Falls die Berechnung eines baryzentrischen Sonnenortes benötigt wird, dann genügt es, die Störungsterme des Mondes aller drei Koordinaten Null zu setzen.

Die Prozedur `Mitt_ScheiOrt` stellt den zweiten Teil des Programmes dar. Die Algorithmen der zu dieser Umrechnung benutzten Programnteile `karpol`, `praezession`, `nutation`, `ekl_schiefe`, `EklSy_AeqSy`, `AeqSy_EklSy`, und `aberration` wurden der Veröffentlichung 5 (BASIC-Programm `KORSA`) entnommen.

### Testbeispiel 0.1.1989

Um die Inbetriebnahme des Programmes zu erleichtern, sei hier ein ausführliches Testbeispiel gegeben. Alle Werte wurden intern mit 16-stelliger Genauigkeit berechnet und werden mit acht Stellen nach dem Komma ausgegeben. Diese Stellenzahl widerspiegelt nicht die Verfahrensgenauigkeit !!

JD: 2447526.50000000  
T: 0.88997947

Test: mittlerer Sonnenort

DLP: -0.00171063 Grad  
LO: 279.64031738 Grad  
M: 356.88908116 Grad

Mx.V:	91.81717877	Grad	mittl. Anom. der Venus
Mx.H:	73.76551199	Grad	des Mars
Mx.J:	46.06422530	Grad	des Jupiter
Mx.S:	182.97157183	Grad	des Saturn
D:	269.32607724	Grad	mittl. Abstand Mond-Sonne
A:	193.28156179	Grad	mittl. Anomalie des Mondes
U:	211.13126066	Grad	mittl. Argument der Mondbreite
DL:	359.89384836	Grad	
RO	-0.00730871	AE	
DL2:	0.00182907	Grad	Störung der Länge durch die Venus
DL4:	-0.00029760	Grad	den Mars
DL5:	-0.00057443	Grad	den Jupiter
DL6:	0.00002679	Grad	den Saturn
DLN:	-0.00190671	Grad	den Mond
DR2:	-1660.63954277	AE	Störung des Radius durch die Venus
DR4:	221.29393219	AE	den Mars
DR5:	4302.04171489	AE	den Jupiter
DR6:	-696.56621092	AE	den Saturn
RM:	-0.00000053	AE	den Mond

Radius: 0.9833156 AE

Mittl. Ort:

Länge: 279 Grad 31' 59.67"      Breite: -0 Grad 0' 0.17"

Rektaszension: 18 Stunden 41 min 29.497 sek

Deklination: -23 Grad 5' 53.23"

**Test: mittlerer-scheinbarer Ort**

Die Präzession bewirkt keine Änderung der Koordinaten, da der mittlere Ort schon auf das Äquinox des Datums berechnet ist. Diese Prozedur ist dennoch enthalten, damit diese Umwandlung ohne Probleme in andere Programme übernommen werden kann.

delta\_psi: 6.68011690"

delta\_eps: 8.02593594"

lab: -0.00568611 Grad

bab: 0.00000000 Grad

Scheinbarer Ort:

Länge: 279 Grad 31' 45.88"      Breite: -0 Grad 0' 8.09"

Rektaszension: 18 Stunden 41 min 28.541 sek

Deklination: -23 Grad 6' 2.11"

Zum Vergleich der scheinbare Ort aus "Astronomicheskij Eshegodnik SSSR na 1989 god", Seite 10:

Rektaszension: 18 Stunden 41 min 28.612 sek

Deklination: -23 Grad 6' 1.93"

# Listing des Pascalprogramms "SUN\_NEWCOMB"

```

program sun_newcomb;
const pi=3.141592653589793e0;
      jd1950=2433282.423e0;
      jd1900=2415020e0;
      rad_je_grd=0.017453292519943292e0;
      grd_je_rad=57.295779513082334e0;
var jd,l,b,r,alfa,delta,alfas,deltas,ls,bs:real;

function sgn(x:real):real;
var sg:real;
begin if x=0e0 then sg:=0e0
      else sg:=x/abs(x);
      sgn:=sg end;

function arcsin(x:real):real;
begin arcsin:=arctan(x/(sqrt(1e0-sqr(x)))) end;

function tan(x:real):real;
begin tan:=sin(x)/cos(x) end;

function mod2pi(winkel:real):real;
var sg:real;
begin sg:=sgn(winkel);winkel:=abs(winkel);
      winkel:=2e0*pi*(winkel/(2e0*pi)-trunc(winkel/(2e0*pi)));
      mod2pi:=sg*winkel end;

procedure sun_new_mitt(jd:real;var l,b,r:real);
const grd_sek=2.7777777777777777e-4;
type ZeitTermtyp=record t,tsq,tcb:real end;
      PlanAnomtyp=record V,M,J,S:real end;
var TT:Zeittermtyp;
      datei:text;
      DLP,M,D,A:real;
      Mx:PlanAnomtyp;

function dsin(X:real):real;begin dsin:=sin(X*rad_je_grd) end;
function dcos(X:real):real;begin dcos:=cos(X*rad_je_grd) end;
function mod360(winkel:real):real;
var sg:real;
begin sg:=sgn(winkel);winkel:=abs(winkel);
      winkel:=360e0*(winkel/360e0-trunc(winkel/360e0));
      if sg<0e0 then winkel:=360e0-winkel; mod360:=winkel end;

procedure T_term1900(jd:real;var TT:ZeitTermtyp);
begin with TT do begin
      T:=(jd-2.415020e6)/36525e0;TSQ:=T*T;TCB:=T*TSQ end end;

function fDLP{langper. Störungen}(TT:Zeittermtyp):real;
begin with TT do
      fDLP:=grd_sek*(1.882e0-1.6e-2*T)*dsin(57.24e0+150.27e0*T)+
            grd_sek*6.4e0*dsin(231.19e0+20.2e0*T)+
            grd_sek*0.266e0*dsin(31.8e0+119e0*T) end;

```

```

function fL0{mittl. Sonnenlänge}(TT:ZeitTermtyp;DLP:real):real;
begin with TT do
  fL0:=mod360(279.6966778e0+36000e0*T+grd_sek*2763.13e0*T+
    grd_sek*1.089e0*TSQ+
    grd_sek*0.0202e0*dsin(315.6e0+893.3e0*T)+DLP) end;

function fM{mittl. Anomalie}(TT:ZeitTermtyp;DLP:real):real;
begin with TT do
  fM:=mod360(358.4758333e0+35999e0*T+grd_sek*179.1e0*T-
    grd_sek*0.54e0*TSQ+DLP) end;

procedure plx{mittl. Anom. VMJS}(TT:ZeitTermtyp;var Mx:PlanAnomTyp);
begin with TT do begin
  Mx.V:=mod360(212.45e0+58517.493e0*T);
  Mx.M:=mod360(319.58e0+19139.977e0*T);
  Mx.J:=mod360(225.28e0+3034.583e0*T+grd_sek*1300e0*dsin(133.775e0+
    39.804e0*T));
  Mx.S:=mod360(175.6e0+1221.794e0*T) end end;

function fD{Sonne-Mond}(TT:ZeitTermtyp):real;
begin fD:=mod360(350.737486e0+445267.114217e0*TT.T) end;

function fA{mittl. An. Mond}(TT:ZeitTermtyp):real;
begin fA:=mod360(296.1046e0+477193.849108e0*TT.T) end;

function fU{Arg. d. Mondbreite}(TT:ZeitTermtyp):real;
begin fU:=mod360(11.25089e0+483202.025150e0*TT.T) end;

function fDL{wahr.-mittl. Sonnenl.}(TT:ZeitTermtyp;M:real):real;
begin fDL:=mod360(grd_sek*(6910.057e0-17.24e0*TT.T)*dsin(M)+
  grd_sek*(72.338e0-0.361e0*TT.T)*dsin(M+M)+
  grd_sek*1.054e0*dsin(M+M+M)) end;

function fRO{ungest. Radiuslog}(TT:ZeitTermtyp;M:real):real;
begin with TT do
  fRO:=3.057e-5-1.5e-7*T+(-7.27412e-3+1.814e-5*T)*dcos(M)+
    (-9.138e-5+4.6e-7*T)*dcos(M+M)-1.45e-6*dcos(M+M+M) end;

function fDLx{Störungen VMJS}
  (TT:ZeitTermtyp;M:real;Mx:PlanAnomTyp):real;
var DL2,DL4,DL5,DL6,DLM:real;FA,WI:real; I,I1,I2:integer;
begin
  DL2:=0e0; for I:=1 to 12 do begin
    read(datei, FA,WI,I1,I2);
    DL2:=DL2+grd_sek*FA*dcos(WI+I1*Mx.V+I2*M) end;
  DL4:=0e0; for I:=1 to 11 do begin
    read(datei, FA,WI,I1,I2);
    DL4:=DL4+grd_sek*FA*dcos(WI+I1*Mx.M+I2*M) end;
  DL5:=0e0; for I:=1 to 8 do begin
    read(datei, FA,WI,I1,I2);
    DL5:=DL5+grd_sek*FA*dcos(WI+I1*Mx.J+I2*M) end;
  DL6:=0e0; for I:=1 to 4 do begin
    read(datei, FA,WI,I1,I2);
    DL6:=DL6+grd_sek*FA*dcos(WI+I1*Mx.S+I2*M) end;
  fDLx:=DL2+DL4+DL5+DL6 end;

```

```

function fDLM(Störungen Mond)(D,A,M:real):real;
begin fDLM:=grd_sek*(6.454e0*dsin(D)+0.177e0*dsin(D+A)-
0.424e0*dsin(D-A)+0.172e0*dsin(D-M)) end;

function fDRx(Störungen)(M:real;Mx:PlanAnontyp):real;
var DR2,DR4,DR5,DR6:real;FA,WI:real;i,I1,I2:integer;
begin
DR2:=0e0; for I:=1 to 11 do begin
read(datei, FA,WI,I1,I2); DR2:=DR2+FA*dcos(WI+I1*Mx.V+I2*M) end;
DR4:=0e0; for I:=1 to 9 do begin
read(datei, FA,WI,I1,I2); DR4:=DR4+FA*dcos(WI+I1*Mx.M+I2*M) end;
DR5:=0e0; for I:=1 to 10 do begin
read(datei, FA,WI,I1,I2); DR5:=DR5+FA*dcos(WI+I1*Mx.J+I2*M) end;
DR6:=0e0; for I:=1 to 3 do begin
read(datei, FA,WI,I1,I2); DR6:=DR6+FA*dcos(WI+I1*Mx.S+I2*M) end;
fDRx:=(DR2+DR4+DR5+DR6)/1e9 end;

function fRM(Störungen Mond)(D,A,M:real):real;
begin fRM:=(13360e0*dcos(D)+370e0*dcos(D+A)-1330e0*dcos(D-A)-
140e0*dcos(D+M)+360e0*dcos(D-M))/1e9 end;

function fR(RO,DRx,RM:real):real;
begin fR:=exp(ln(10e0)*(RO+DRx+RM)) end;

function fB(M,U:real;Mx:PlanAnontyp):real;
var FA,WI:real;I1,I2:integer;B:real;
begin B:=0e0;
read(datei, FA,WI,I1,I2); B:=B+FA*dcos(WI+I1*Mx.V+I2*M);
read(datei, FA,WI,I1,I2); B:=B+FA*dcos(WI+I1*Mx.J+I2*M);
read(datei, FA); B:=B+FA*dsin(U); fB:=grd_sek*B end;

function fL(LO,DL,DLx,DLM:real):real;
begin fL:=mod360(LO+DL+DLx+DLM) end;

begin{sun_new_mitt}
assign(datei,'a:sunnew.dat');reset(datei);
T term1900(jd, TT);
DLP:= fDLP(TT);
M:=fM(TT,DLP);
pMx(TT, Mx);
D:=fD(TT);
A:=fA(TT);
L:=fL(fLO(TT,DLP),fBL(TT,M),fDLx(TT,M,Mx),fDLM(D,A,M))*rad_je_grd;
R:=fR(fRO(TT,M),fDRx(M,Mx),fRM(D,A,M));
B:=fB(M,fU(TT),Mx)*rad_je_grd;
end;{sun_new_mitt}

procedure karpol(x,y:real;var winkel,radius:real);
begin radius:=sqrt(sqr(x)+sqr(y));
if (x=0e0) or (y=0e0) then
winkel:=sgn(y)*pi/2+sgn(x)*(sgn(x*pi/2-pi/2)
else winkel:=arctan(y/x)-(sgn(x)-1e0)*sgn(y)*pi/2 end;

```

```

procedure praezession(jdin, jdout, alfa, delta:real;
                    var alfaout, deltaout:real);
var rad_je_sek, tau0, tau, zeta, kappa, theta, A, B, C, radius, winkel:real;
begin
  rad_je_sek:=rad_je_grd/3600e0;
  tau0:=(jdin-2415020.313e0)/36524.2199e0;
  tau:=(jdout-jdin)/36524.2199e0;
  zeta:=rad_je_sek*((2304.25e0+1.396e0*tau0)*tau+
                    0.302e0*tau*tau+0.018e0*tau*tau*tau);
  kappa:=zeta+rad_je_sek*(0.791e0*tau*tau+0.001e0*tau*tau*tau);
  theta:=rad_je_sek*((2004.682e0-0.853e0*tau0)*tau-
                    0.426e0*tau*tau-0.042e0*tau*tau*tau);
  A:=cos(delta)*sin(alfa+zeta);
  B:=cos(theta)*cos(delta)*cos(alfa+zeta)-sin(theta)*sin(delta);
  C:=sin(theta)*cos(delta)*cos(alfa+zeta)+cos(theta)*sin(delta);
  karpol(B, A, winkel, radius);
  alfaout:=winkel+kappa;
  deltaout:=arcsin(C) end;

```

```

procedure nutation(jd:real; var delta_psi, delta_eps:real);
var T, TSQ, L, Lstrich, M, Mstrich, omega:real;
begin T:=(jd-jd1900)/36525e0; TSQ:=T*T;
  L:=rad_je_grd*(279.6967e0+36000.7689e0*T+0.000303e0*TSQ);
  Lstrich:=rad_je_grd*(270.4342e0+431267.8831e0*T-0.001133e0*TSQ);
  M:=rad_je_grd*(358.4758e0+35999.0498e0*T-0.00015e0*TSQ);
  Mstrich:=rad_je_grd*(296.1046e0+477198.8491e0*T+0.009192e0*TSQ);
  omega:=rad_je_grd*(259.1833e0-1934.1420e0*T+0.002078e0*TSQ);
  delta_psi:=-(17.2327e0+0.01737e0*T)*sin(omega)-
    (1.2729e0+0.00013e0*T)*sin(L+L)
    +0.2088e0*sin(omega+omega)-0.2037e0*sin(Lstrich+Lstrich)
    +(0.1261e0-0.00031e0*T)*sin(M)+0.0675e0*sin(Mstrich)
    -(0.0497e0-0.00012e0*T)*sin(L+L+M)
    -0.0342e0*sin(Lstrich+Lstrich-omega)
    -0.0261e0*sin(Lstrich+Lstrich+Mstrich)
    +0.0214e0*sin(L+L-M)-0.0149e0*sin(L+L-Lstrich-Lstrich+Mstrich)
    +0.0124e0*sin(L+L-omega)+0.0114e0*sin(Lstrich+Lstrich-Mstrich);
  delta_psi:=delta_psi/3600e0*rad_je_grd;
  delta_eps:=(9.21+0.00091e0*T)*cos(omega)
    +(0.5522e0-0.00029e0*T)*cos(L+L)
    -0.0904e0*cos(omega+omega)+0.0884e0*cos(Lstrich+Lstrich)
    +0.0216e0*cos(L+L+M)+0.0183e0*cos(Lstrich+Lstrich-omega)
    +0.0113e0*cos(Lstrich+Lstrich+Mstrich)-0.0093e0*cos(L+L-M)
    -0.0066e0*cos(L+L-omega);
  delta_eps:=delta_eps/3600e0*rad_je_grd end;

```

```

function ekl_schiefe(jd:real):real;
var T:real;
begin T:=(jd-jd1900)/36525e0;
  ekl_schiefe:=(23.452294e0-0.0130125e0*T-0.00000164e0*sqr(T)+
    0.000000503e0*T*sqr(T))*rad_je_grd end;

```

```

procedure EklSy_AeqSy(l, b, jd:real; var alfa, delta:real);
var SE, CSE, SSE, radius:real;
begin SE:=ekl_schiefe(jd); CSE:=cos(SE); SSE:=sin(SE);
  delta:=arcsin(CSE*sin(b)+SSE*cos(b)*sin(l));
  karpol(cos(l), sin(l)*CSE-tan(b)*SSE, alfa, radius) end;

```

```

procedure AeqSy_EklSy(alfa,delta,jd:real; var l,b:real);
var SE,CSE,SSE,radius:real;
begin SE:=ekl_schiefe(jd);CSE:=cos(SE);SSE:=sin(SE);
      b:=arcsin(CSE*sin(delta)-SSE*cos(delta)*sin(alfa));
      karpol(cos(alfa),sin(alfa)*CSE+tan(delta)*SSE, l,radius) end;

procedure aberration(l,b,jd:real; var lab,bab:real);
var ab_konst,LS:real;
function sonnenlaenge(jd:real):real;
var T,L,M,C:real;
begin T:=(jd-jd1900)/36525e0;
      L:=(279.69668e0+36000.76892e0*T+0.0003025e0*T*T)*rad_je_grd;
      M:=(358.47583e0+35999.04975e0*T-0.00015e0*T*T-
          0.0000033e0*T*T*T)*rad_je_grd;M:=mod2pi(M);
      C:=((1.91946e0-0.004789e0*T-0.000014e0*T*T)*sin(M)+
          (0.020094e0-0.0001e0*T)*sin(M+M)+
          0.000293e0*sin(M+M+M))*rad_je_grd;
      sonnenlaenge:=mod2pi(L+C) end;
begin LS:=sonnenlaenge(jd);
      ab_konst:=20.47e0/3600e0*rad_je_grd;
      lab:=-ab_konst/cos(b)*cos(l-LS);
      bab:=+ab_konst*sin(b)*sin(l-LS) end;

procedure Mitt_ScheiOrt(alfa,delta,jdstart,jd:real;var
alfas,deltas:real);
var al,de,delta_psi,delta_eps,SAL,CAL,TDE,ES,SES,CES,lab,bab:real;
begin
      praezession(jdstart,jd,alfa,delta, al,de);
      nutation(jd, delta_psi,delta_eps);
      SAL:=sin(al);CAL:=cos(al);TDE:=tan(de);
      ES:=ekl_schiefe(jd);SES:=sin(ES);CES:=cos(ES);
      al:=al+(delta_psi*(CES+SES*SAL*TDE)-delta_eps*(CAL*TDE));
      de:=de+(delta_psi*(SES*CAL)+delta_eps*SAL);
      AeqSy_EklSy(al,de,jd, l,b);
      aberration(l,b,jd, lab,bab);
      EklSy_AeqSy(l+lab,b+bab,jd, alfas,deltas);
      alfas:=mod2pi(alfas) end;

procedure wrwinkel(wi:real);
var grd,min:integer;sec:real;vz:boolean;
begin vz:=sgn(wi)<>-1e0;wi:=abs(wi);
      wi:=grd_je_rad*wi;
      grd:=trunc(wi);wi:=60e0*(wi-grd);
      min:=trunc(wi);sec:=60e0*(wi-min);
      if vz then write(' ') else write('-');
      write(grd:3,' Grad ',min:2,chr(39),' ',sec:5:2,'") end;

procedure wrzeit(wi:real);
var std,min:integer;sec:real;vz:boolean;
begin vz:=sgn(wi)<>-1e0;wi:=abs(wi);
      wi:=grd_je_rad/15e0*wi;
      std:=trunc(wi);wi:=60e0*(wi-std);
      min:=trunc(wi);sec:=60e0*(wi-min);
      if vz then write(' ') else write('-');
      write(std:2,' Std ',min:2,' min ',sec:6:3,' sek') end;

```



```

begin{main}
  writeln ('SUN_NEWCOMB (mittl. Aequinox des Datums): piu - 8.8.88');
  write('JD:');readln(jd);
  sun_new_mitt(jd, l, b, r);
  EklSy_AeqSy(l, b, jd, alfa, delta);
  writeln('Radius:', r:10:7, ' AE');
  writeln('Mittl. Ort');
  write('Länge:');wrwinkel(l);write(' ');
  write('Breite:');wrwinkel(b);writeln;
  if alfa<0 then alfa:=alfa+pi+pi;
  write('Rektaszension:');wrzeit(alfa);write(' ');
  write('Deklination:');wrwinkel(delta);writeln;
  Mitt_ScheiOrt(alfa, delta, jd, jd, alfas, deltas);
  AeqSy_EklSy(alfas, deltas, jd, ls, bs);
  if ls<0e0 then ls:=-ls+pi+pi;
  writeln('Scheinbarer Ort:');
  write('Länge:');wrwinkel(ls);write(' ');
  write('Breite:');wrwinkel(bs);writeln;
  if alfas<0 then alfas:=alfas+pi+pi;
  write('Rektaszension:');wrzeit(alfas);write(' ');
  write('Deklination:');wrwinkel(deltas);writeln;
  writeln('SUN_NEWCOMB beendet.') end.

```

Inhalt der Datei "SUNNEW.DAT

4.833	299.102	1	-1
0.116	148.9	2	-1
5.526	148.313	2	-2
2.497	315.943	2	-3
0.666	177.71	3	-3
1.559	345.253	3	-4
1.024	318.15	3	-5
0.21	206.2	4	-4
0.144	195.4	4	-5
0.152	343.8	4	-6
0.123	195.3	5	-7
0.154	<del>356.6</del>	5	-8

*559.6*

0.273	217.7	-1	1
2.043	343.888	-2	2
1.77	200.402	-2	1
0.129	294.2	-3	3
0.425	338.88	-3	2
0.5	105.18	-4	3
0.585	334.06	-4	2
0.204	100.8	-5	3
0.154	227.4	-6	4
0.101	96.3	-6	3
0.106	222.7	-7	4

0.163	198.6	-1	2
7.208	179.532	-1	1
2.6	263.217	-1	0
2.731	87.145	-2	2
1.61	109.493	-2	1
0.164	170.5	-3	3
0.556	82.65	-3	2
0.21	98.5	-3	1

0.419	100.58	-1	1
0.32	269.46	-1	0
0.108	290.6	-2	2
0.112	293.6	-2	1

2359	209.03	1	-1
160	58.4	2	-1
5842	58.318	2	-2
869	226.7	2	-3
1045	87.57	3	-3
1497	255.25	3	-4
194	49.5	3	-5
376	116.28	4	-4
196	105.2	4	-5
163	145.4	5	-5
141	105.4	5	-7

150	127.7	-1	1
2057	253.828	-2	2
151	295	-2	1
168	203.5	-3	3
215	249	-3	2
478	15.17	-4	3
105	65.9	-4	2
107	324.6	-5	4
139	137.3	-6	4

208	112	-1	2
7067	89.545	-1	1
244	338.6	-1	0
103	350.5	-2	3
4026	357.108	-2	2
1459	19.467	-2	1
281	81.2	-3	3
803	352.56	-3	2
174	8.6	-3	1
113	347.7	-4	2

429	10.6	-1	1
162	200.6	-2	2
112	203.1	-2	1

-0.21	151.8	3	-4
-0.166	265.5	-2	1
0.576			

Autor:

Uwe Pilz  
Lazarusstr.23  
Leipzig  
7024

Arbeitskreis "Numerische Astronomie"  
PST 29  
Schkeuditz  
7144